

---

Subject: Custom Control

Posted by [jeremy\\_c](#) on Sat, 24 Jul 2010 16:18:17 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

I've created a custom control and have the chicken/egg problem I think. My new control uses a layout. Thus, I have created a new layout in my main applications .lay file named DxccLayout.

I then created DxccWidget.h and DxccWidget.cpp files. The .h file currently looks like:

```
#ifndef _SmartLog_DxccWidget_h_
#define _SmartLog_DxccWidget_h_

class DxccWidget : public WithDxccWidgetLayout<ParentCtrl>
{
private:

public:
    typedef DxccWidget CLASSNAME;

    DxccWidget();
};

#endif
```

My main window uses this widget. So, the problem is that the layout file will not compile because it does not know about the custom control DxccWidget. When I include DxccWidget.h before the layout file, it fails to compile as it will not compile without knowing about the DxccLayout defined in the .lay file.

How is this normally handled?

Jeremy

---

---

Subject: Re: Custom Control

Posted by [koldo](#) on Sun, 25 Jul 2010 06:52:04 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hello Jeremy

I am not fully sure about your question.

If I need to include in a .lay file class a ctrl designed in another .lay file, I do this:

Imagine in Small.lay file it is designed a ctrl used in Big.lay, so, Package.h file contains:

```
#define LAYOUTFILE <MyProject/Small.lay>
#include <CtrlCore/lay.h>

struct SmallClass : WithSmallClass<ParentCtrl> {
    typedef SmallClass CLASSNAME;

    SmallClass();
    ...
};

#define LAYOUTFILE <MyProject/Big.lay>
#include <CtrlCore/lay.h>

struct BigClass : WithBigClass<ParentCtrl> {
    typedef BigClass CLASSNAME;

    BigClass();
    ...
};
```

---

Subject: Re: Custom Control  
Posted by [jeremy\\_c](#) on Sun, 25 Jul 2010 19:03:13 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thanks. I didn't realize I'd have to create different layouts, however, that's fine. Easy enough to do and decouples the control from the app a bit.

Jeremy